Please type a plus sign (+) inside this box → [+]

PTO/SB/21 (08-00)
Approved for use through 10/31/2002. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	09/096,939
Filing Date	June 12, 1998
First Named Inventor	Kult
Group Art Unit	2743
Examiner Name	Barnie,
Total Number of Pages in This Submission: 72	Attorney Docket Number: CDR-97-031

RECEIVED
OCT 10 2001
Technology Center 2600

ENCLOSURES *(check all that apply)*

- [ ] Fee Transmittal Form
  - [ ] Fee Attached
- [ ] Amendment / Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts/ Incomplete Application
  - [ ] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Assignment Papers *(for an Application)*
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance Communication to Group
- [x] Appeal Communication to Board of Appeals and Interferences
- [ ] Appeal Communication to Group *(Appeal Notice, Brief, Reply Brief)*
- [ ] Proprietary Information
- [ ] Status Letter
- [x] Other Enclosure(s) *(please identify below)*: Return Receipt Postcards (2)

Remarks

BOARD OF PATENT APPEALS AND INTERFERENCES
OCT 09 2001
RECEIVED

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	Rudolph J. Buchel, Jr. JONES, DAY, REAVIS & POGUE
Signature	Rudolph J. Buchel
Date	October 1, 2001

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, Washington, DC 20231 on this date: Oct. 1, 2001

Typed or printed name	Marsha S. Kappus		
Signature	Marsha S. Kappus	Date	October 1, 2001

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

#13

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application: **Kult**	§	
Serial No.: **09/096,939**	§	Group Art Unit: **2743**
Filed: **June 12, 1998**	§	Examiner: **Barnie, R.**
For: **A SYSTEM AND METHOD FOR RESOURCE MANAGEMENT**	§	Attorney Docket No.: **CDR-97-031**

Certificate of Mailing Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being deposited with the United States Postal Service as First Class mail in an envelope addressed to: Assistant Commissioner of Patents, Washington, D.C. 20231 on October 1, 2001.

By: Marsha S. Kappus
Marsha S. Kappus

Assistant Commissioner for Patents
Washington, D.C. 20231

ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S BRIEF (37 C.F.R. 1.192)

This brief is in furtherance of the Notice of Appeal, filed in this case on **January 12, 2001.**

The fees required under § 1.17(c), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate. (37 C.F.R. 1.192(a))

This brief contains these items under the following headings, and in the order set forth below (37 C.F.R. 1.192(c)):

I. REAL PARTIES IN INTEREST (37 C.F.R. 1.192(c)(1))

The real party in interest in this appeal is the following party: WorldCom, Inc.

II. RELATED APPEALS AND INTERFERENCES (37 C.F.R. 1.192(c)(2))

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

III. STATUS OF CLAIMS (37 C.F.R. 1.192(c)(3))

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 20

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: None
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: **1 - 20**
4. Claims allowed: None
5. Claims rejected: **1 - 20**

C. CLAIMS ON APPEAL

The claims on appeal are: **1 - 20**

IV. STATUS OF AMENDMENTS (37 C.F.R. 1.192(c)(4))

An after final amendment was filed but not entered by the Examiner.

V. SUMMARY OF INVENTION (37 C.F.R. 1.192(c)(5))

The present invention is directed to managing resources in a telecommunication network that are related to or accessible by a switch, such as Intelligent Service Networks (ISNs) **102**

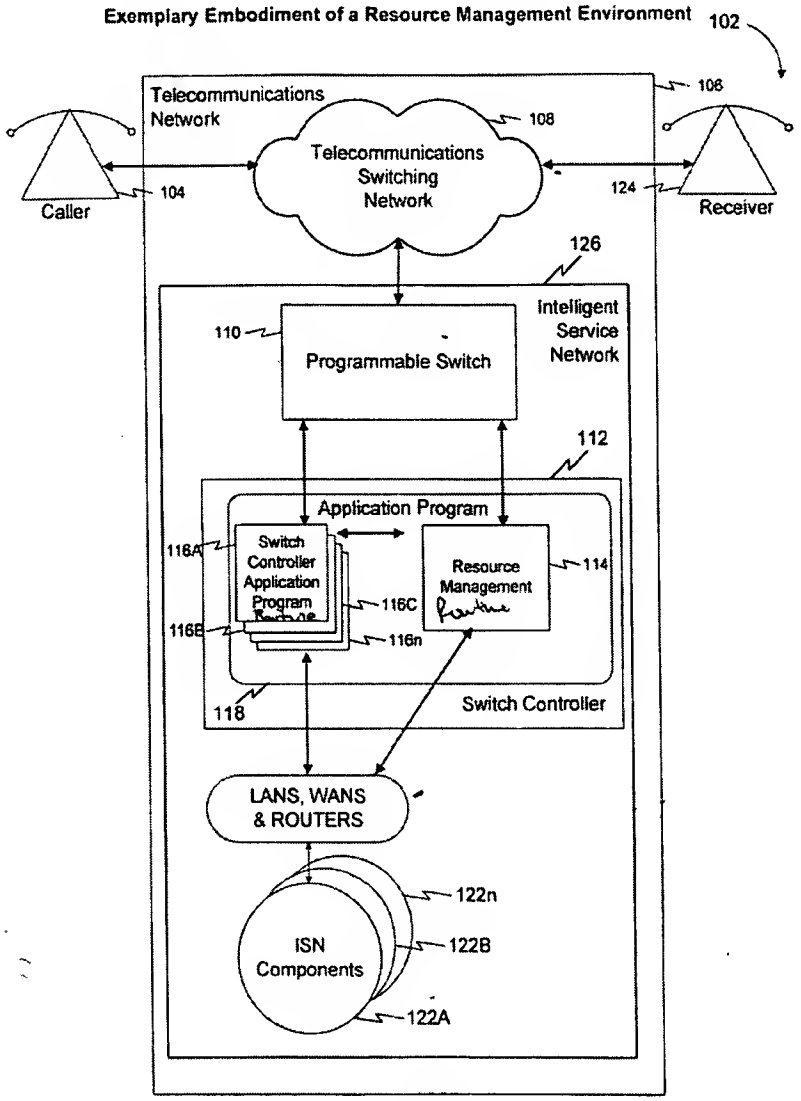


FIG. 1

shown in **FIG. 1**, reproduced herein for convenience.

Prior art telecommunications networks require that, for each unique resource, a requestor understands and utilizes particular Application Programmer Interfaces (APIs) in order for the requestor to access or update the information related to the resource. Typically, all resource information is stored and managed by the particular resource. Thus, each time a requestor

desires resource information or to update resource information for a particular resource, according to the prior art, the requestor must express a query using an API understood by the particular resource and then transmit the query to the resource itself for processing. These prior art APIs are resource-based.

By contrast, the present invention uses a resource management routine which contains individual resource manager correlating to the resources being managed by the resource management routine. Each resource manage comprises a table of data related to the resource to which it corresponds and to a generic resource management API for acquiring data from the table. All of the generic resource management APIs are standard resource APIs enabling the requestor to query a resource without need knowing each APIs that is understood by each particular resource (see page 3, line 22 to 5, line 10).

Resource management routine **114** managing all queries for resources associated with, for

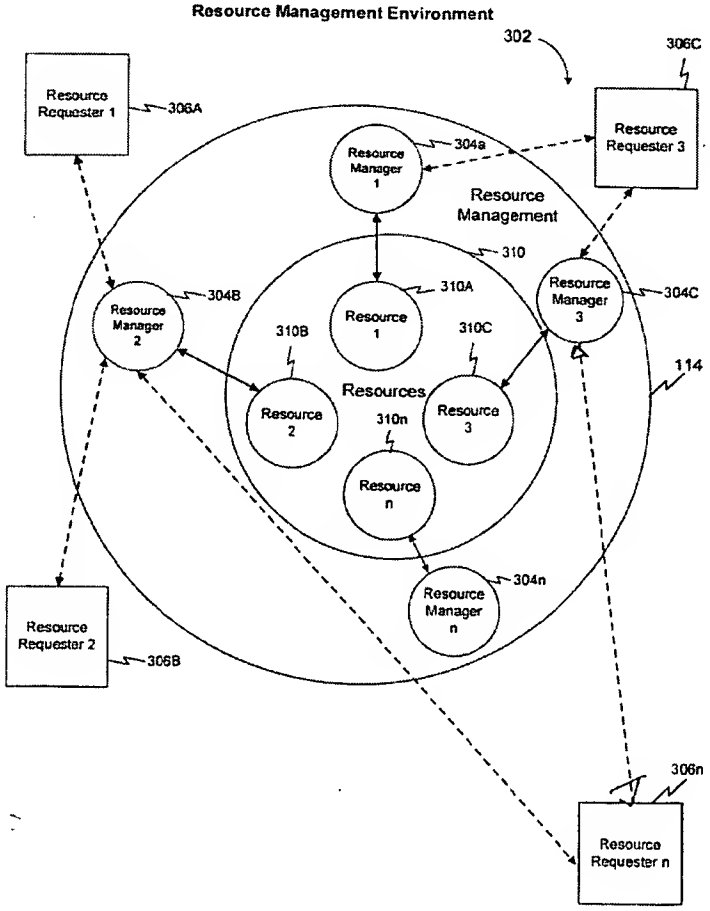


FIG. 3

instance, an ISN. When used in ISN **116**, resource management routine **114** resides in the memory of switch controller **112**.

With regard to **FIG. 3**, resource management routine **114** uses individual resource managers **304A - 304n** to handle queries originating from any of requestors **306A - 306n** directed to respective one of corresponding resources **310A - 310n**. Resources **310A - 310n** may be of different type categories (*i.e.,* internal operational resources, external components, or applications processing data). With the inclusion of resource managers **304A - 304n** (within resource management routine **114**), requestors **306A - 306** do not interface directly with resources **310A - 310n** instead each of resource managers **304A - 304n** contains the resource information and procedures necessary for interfacing and obtaining information from the respective resources **310A - 310n**. In this case, requestors **306A - 306** actually interface with a particular resource manager **304** associated with a particular resource **310** that requester **306** intends to interface with. Thus, it can be seen that resource management routine **114** serves as a protective layer between requestors **306A - 306n** and resources **310A - 310n** as is more clearly illustrated in **FIG. 3**. However, in practice an individual resource manager actually interface between its corresponding resource and any of requestor needing to interface with the resource, as also can be appreciated from **FIG. 3** but is more apparent from **FIG. 4**.

According to the present invention, each of resource managers **304A - 304n** serve two roles, they: store information about a particular corresponding resource **310** (in resource manager table **406**); and provide a standard mechanism for all requestors **306** to interface with and access information associate with that resource **310** (by resource manager API **404**). Each resource manager **304** maintains resource information about a particular corresponding resource **310** in tabular form as a resource manager table **406** in a memory, for instance in memory **208** associated with switch controller **112**. Requestors **306** query a particular resource manager **304** using that manager's resource manager API **404** for that information. The resource manager **304** being queried uses information contained in the structure of the query from requestor **306** for retrieving specified data from resource manager table **406** and/or transacting with resource **310** (see page 18, line 9 to page 19, line 14). In sharp contrast with the prior art, resource manager APIs **404** are generic and not specialized APIs, specific to a unique resource. Requestor **306** does not directly communicate with any of resources **310A - 310n**, but instead interfaces with

resource managers **304A - 304n** contained in resource management **114**. Thus, requestors **306** need only understand the standardized generic APIs common to all to resource managers **304A - 304n** in order to access all resources' data managed by resource management routine **114.**

In accordance with one exemplary embodiment of the present invention, each unique resource information table **406** corresponds to a particular resource **310** and is stored in the

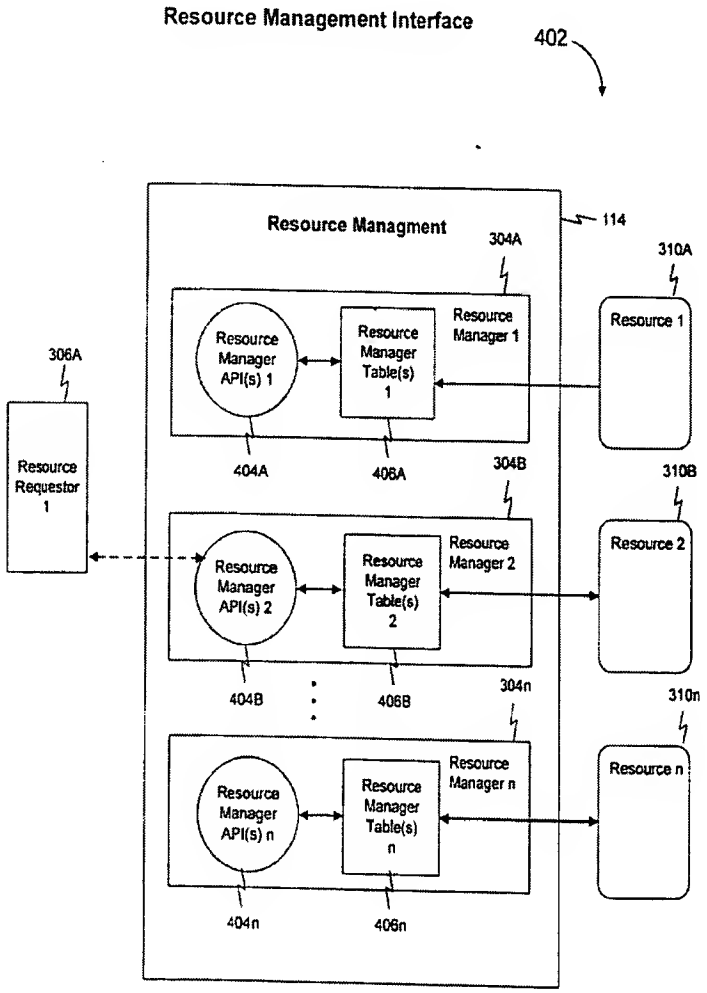


FIG. 4

switch controller's memory, for instance memory **118**. Similarly, every resource management API **404** corresponds to particular resource **310** and therefore, provides a unique table of information **406** for particular resource **310** is also stored in the switch controller's memory.

VI. ISSUES (37 C.F.R. 1.192(c)(6))

The issues on appeal are whether:

(1) Claim **5** is unpatentable by virtue of being indefinite under the meaning of 35 U.S.C. § 112 second paragraph;

(2) Claims **1 - 8, 10, 11, 15, 16, 18** and **19** are unpatentable over Sofman (U.S. Patent No. 5,937,042) under 35 U.S.C. § 102(e).

(3) Claim **9** is unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Taylor, et al. (U.S. Patent No. 5,912,961) under 35 U.S.C. § 103(a).

(4) Claims **12 - 14** and **17** are unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Gottlieb (U.S. Patent No. 5,920,621) under 35 U.S.C. § 103(a).

(5) Claim **20** is unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Reto et al. (U.S. Patent No. 5,825,857) under 35 U.S.C. § 103(a).

VII. GROUPING OF CLAIMS (37 C.F.R. 1.192(c)(7))

The claims do not stand or fall together as a single group. Instead, the claims stand and fall in three groups as follows:

Group A -- claims **1, 4** and **19;**

Group B -- claims **2, 6** and **7;**

Group C -- claims **3, 8, 10, 11, 15, 16** and **18**;

Group D -- claim **5;**

Group E -- claim **9**;

Group F -- claims **12 - 14** and **17**; and

Group G -- claim **20.**

VIII.A. ARGUMENTS—REJECTIONS UNDER 35 U.S.C. 112, (37 C.F.R. 1.192(c)(8)(i))

I. 35 U.S.C. § 112, Second Paragraph

The Examiner has rejected claim no. **5** under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter, which applicants regard as the invention. This rejection should not be upheld.

It should be readily apparent from the present specification, the resource management means of the present invention may be comprised of a plurality of independent resource managers. Claim **1** recites a "resource management means comprises one or more resource managers, said resource managers being one of:" semaphore resource manager; switch controller resource manager; agent resource manager; call data block resource manager; service logic resource manager; **or** switch resource resource manager. In addition to one or more of the above recited resource managers, claim **5** recites that the resource management means also includes a table manager resource manager; a queue logic manager resource manager; a system manager resource manager; **and** a shared memory manager resource manager. Each of these particular types of resource managers is shown on **FIG. 6** and described on pages 22 - 26 as being part of the resource management means.

It is not at all clear what the Examiner feels is indefinite about either claim **1** or claim **5** or the combination of claims. Claim **1** merely lists a plurality of unique resource manager types in the alternative, thus claim **1** allows for the possibility that the resource management means comprises only one resource manager. By contrast, claim **5** lists another plurality of unique resource managers types; however, the listing is cumulative and not alternative, thus claim **5** requires that the resource management means comprise more than one resource manager. Appellant's representative can see no possible indefinite language or even ambiguous claim language, especially when given the specificity of the present specification.

Therefore, the rejection of claim no. **5** under 35 U.S.C. § 112, second paragraph, should not be upheld.

VIII.B. ARGUMENTS—REJECTIONS UNDER 35 U.S.C. 103, (37 C.F.R. 1.192(c)(8)(i))

I. The rejection of claims 1, 4 and 19 under U.S.C. § 102 is incorrect.

The Examiner has rejected claim nos. **1-8, 10-11, 15-16** and **18-19** under 35 U.S.C. § 102(e) as being anticipated by Sofman (U.S. Patent No. 5,937,042).

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. In re Bond, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). In addition, all limitations of the claimed invention must be considered when determining patentability. In re Lowry, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. Kalman v. Kimberly-Clark Corp., 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983).

The rejections and discussion provided by the Examiner are vague and sometimes contradictory requiring that this appeal, as were the responses to the Examiner's Office Action, to assume the best possible correlation to the claim limitation and then distinguish over those correlations.

a. With respect to claims **1, 4** and **19,** nowhere does the prior art cited by the Examiner describe **a resource management means for enabling said processor to provide standardized management of multiple resources including internal operational resources, external components, and applications processing data** as required by claim **1**.

Initially, on page 3, paragraph 4, the Examiner states "Sofman teaches ... comprising of processor (204), a resource management means (*see column* 7 lines 7-8 and 23-24) which can be used stored with data storage means." Apparent here the Examiner contends that Sofman's description of a network management application (NMA) anticipates the resource management means recited in claim **1**. However, network management products are well known in the art as

admitted by Sofman. These products are useful for, *inter alia*, discovering, mapping and viewing networks, as well as viewing network device data from devices connected to the visible networks. Nowhere does Sofman describe a network management application that **enables a processor to provide standardized management of multiple resources** as required by claim **1**, nor does the Examiner point to such a limitation in Sofman.

Furthermore, nowhere does Sofman describe a network management application that provides for **standardized management internal operational resources, external components, and applications processing data** as required by claim **1**. At best, Sofman describes a network management application that gathers data from external components.

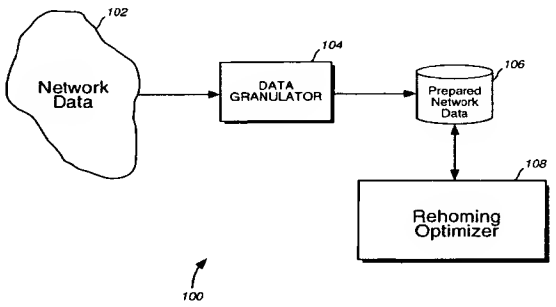


FIG. 1

While much of the Examiner's rationale is extremely hard to follow, apparently after stating that the Sofman's network management application anticipates the resource management means, the Examiner then contradicts his analysis and then infers that somehow data granulator **104** and/or rehoming optimizer **108** act as a **resource management means.** With regard to the teaching of Sofman, as depicted in **FIGs. 1** and **2A**, Sofman describes a system and method for rehome optimization. Essentially, rehome optimization involves acquiring network data, storing it in a separate database as switch and rehome circuit group (RCG) data and then optimizing the data based on user-specified constraints. Data granulator **104** accesses network data **102** and builds switch and RCG data into a Database **106**. Rehoming Optimizer **108** uses database **106** as input. Rehoming optimizer **108** interfaces with a user through an EUI. The user specifies cost objectives and constraints for focusing rehoming optimizer calculations which are used by rehoming optimizer **108**, together with data from prepared network database **106**, to calculate rehoming solutions for optimal network configurations. Solutions are then presented and managed through the EUI **114** of the rehoming optimizer. Nowhere does Sofman describe that the optimization results are returned to prepared network database **106.**

Importantly, Sofman describes data granulator **104** and rehoming optimizer **108** as independent processes synchronized to established copies of data in prepared network database **106** and data granulator **104** executes such that prepared network database **106** will contain a substantially recent snapshot of data for use by the rehoming optimizer **108**. However, at no time does the requestor or user utilize data granulator **104** for accessing network data from database **106**. Instead, the user accesses the data by using rehoming optimizer **108** via EUI driver **114**.

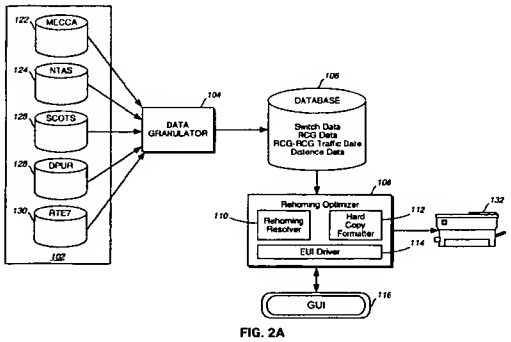


FIG. 2A

Here again, Sofman does not teach that either data granulator **104** or rehoming optimizer **108 enables a processor to provide standardized management of multiple resources** as required by claim **1.** Sofman teaches no more than the admitted prior art, *i.e.* that communication with the resources requires that the requestor use an API that the resource can understand, a resource-based API. Moreover, even if data granulator **104** retrieves the data from network **102** in response to a query from a requestor, a feature limitation definitely not taught or suggested by Sofman, nowhere does Sofman teach or suggest that data granulator **104** comprises any type of API whatsoever.

Furthermore, assuming *arguendo*, that an API of some type was used and that network data **102** could even be considered a resource, at best network data **102** could only be characterized as a data resource for external components. Thus, Sofman cannot teach **standardized management of internal operational resources, external components, and applications processing data** as required by claim **1**, because even if the network data is considered a resource, it is related only to external components.

Because no reference teaches or suggests **a resource management means for enabling said processor to provide standardized management of multiple resources including internal operational resources, external components, and applications processing data** as required by claim **1**, the Examiner has not met the burden, and it is respectfully urged that the Examiner's rejection of claims **1, 4** and **19** should not be sustained.

b. In addition, nowhere does the prior art cited by the Examiner describe that a **resource managers comprises:**

one or more resource manager application program interfaces that manage said internal operational resources, said external components, and said applications processing data; and

one or more data storing means for enabling said processor to store data in table format related to said internal operational resources, said external components, and said applications processing data, said application interfaces manipulating the data to reflect the current resource state as required by claim **1** nor does the Examiner contend that each limitation is taught by Sofman.

Assuming, *arguendo*, that Sofman's data granulator **104**, prepared network database **106** and rehoming optimizer **108** are somehow equivalent to the **resource management means** of claim **1**, *although not directly alleged by the Examiner*, and data granulator **104** corresponded to a **resource manager**, Sofman's fails to teach that data granulator **104**:

(1) contains a **resource manager application program interfaces that manage said internal operational resources** for any type of resource;

(2) contains **one or more data storing means**;

(3) stores information in any storage **related to said internal operational resources, said external components, and said applications processing data;** or

(4) that any API, much less a resource manager application program interface, provides the specific function of **manipulating the data to reflect the current resource state**.

(1) Nowhere does Sofman teach or suggest that data granulator **104** contains any type of API. Assuming *arguendo* that data granulator **104** even uses an API, Sofman teaches no more than the admitted prior art, in that a resource API is accessed by the processor whenever data granulator **104** requires new information from network data **102**. Nowhere does Sofman teach or suggest that data granulator **104** contains one or more **resource manager application program interfaces** as required by claim **1**.

(2) Furthermore, after data retrieval the network data is stored in a database that is external to network data **102**. *i.e.*, prepared network database **106.** Prepared network database **106** is clearly external to data granulator **104** because rehoming optimizer **108** accesses prepared

network database **106** independently of data granulator **104**. If prepared network database **106** were contained within data granulator **104,** then it would be necessary for rehoming optimizer **108** to access prepared network database **106** via data granulator **104**. However, Sofman teaches the opposite. Sofman teaches, and is an important aspect of the rehoming optimization, that rehoming optimizer **108** and data granulator **104** access prepared network database **106** independently (see col. 4, lines 18- 39, especially lines 30 - 36). Clearly, Sofman teaches away from a resource manager comprising a **data storing means for enabling said processor to store data in table format related to said internal operational resources, said external components, and said applications processing data,** because Sofman teaches that rehoming optimizer **108** must access the data within prepared network database **106** independently from rehoming optimizer **108**. This principle is clearly illustrated in each of **FIGs. 1, 2A** and **2B**. Thus, even if data granulator **104** could be considered a resource manager, data granulator **104** does not contain **one or more data storing means,** but instead it is necessary for the storage means to be independent from prepared network database **106**, and therefore not teach an inclusive storage means as required by claim **1**

(3) Here again, if network data **102** is to be considerd a resource, *although not directly alleged by the Examiner*, it could only be considered a data resource associated with external component and NOT **related to said internal operational resources** or **said applications processing data** as required by claim **1.** Nowhere does Sofman describe any more, nor has the Examiner pointed to such a teaching.

(4) Sofman does not expressly teach that any API, much less a resource manager application program interface, provides the specific function of **manipulating the data to reflect the current resource state**. Assuming, again, that network data **102** is to be considered a resource, Sofamn does not teach the standardized management of multiple resources by data granulator **104** which contains any type of API. If an API is used at all, Sofman teaches no more than the admitted prior art, a resource-based API that may be accessed by the processor whenever data granulator **104** (the requestor) requires new information from network data **102**. At best, the internal functionality of data granulator **104** manipulates (col. 4, lines 36 - 39) data into a recent snapshot of network data for rehoming optimizer **108** and not an API that accesses data from network data **102**. Thus, Sofman cannot teach that one or more **resource manager**

application program interfaces manipulate **the data to reflect the current resource state** as required by claim **1**.

Sofman is simply not concerned with providing a generic API type for all resources that is standardized between all resource managers which enable a requestor to access data and interface associated multiple resources as is the direction of the present invention and required by the claims. Any resource management provided by Sofman is based purely on prior art techniques using resource APIs and not based on generic resource manager APIs that are standardized for all resources managed by a resource management means as required by the claims.

Because no reference teaches or suggests that a **resource managers comprises: one or more resource manager application program interfaces that manage said internal operational resources, said external components, and said applications processing data; and one or more data storing means for enabling said processor to store data in table format related to said internal operational resources, said external components, and said applications processing data, said application interfaces manipulating the data to reflect the current resource state**, the Examiner has not met the burden, and it is respectfully urged that the Examiner's rejection of claims **1, 4** and **19** should not be sustained.

II. The rejection of claims 2, 6 and 7 under U.S.C. § 102 is incorrect.

In addition to the forgoing:

a. Sofman does not teach **... sending a query to a resource manager, wherein said resource manager manages information corresponding to a resource, said resource manager complying with a common standard for resource managers within the network** as required by claim **2.**

First, assuming that the user somehow queries prepared network database **106** via rehoming optimizer **108**, then data granulator **104** is never included in the process. Thus, Sofman cannot teach **sending a query to a resource manager** when data granulator **104** is alleged to be the resource manager as required by claim **2** and *apparently relied on by the Examiner*. If, on the other hand, and **nowhere contended by the Examiner**, rehoming optimizer **108** is the resource manager, then rehoming optimizer **108** cannot **resource manager**

complying with a common standard for resource managers within the network as required by claim **2**, because Sofman describes a network where rehoming optimizer **108 MUST** be the sole rehoming optimizer (resource manager) in the network. Thus, no **common standard for resource managers within the network** can exist in Sofman's network as there is only one possible resource manager. Nor is there a need or motivation for a **resource manager complying with a common standard for resource managers within the network** in situations where all queries are handled by the same resource manager.

Moreover, at best rehoming optimizer **108** manages information from prepared network database **106** corresponding to data granulator **104**, and does not manage information that corresponds to any network resource. It is important that rehoming optimizer **108** see a snapshot of the network and not merely information related to a single resource. Thus, Sofman cannot teach, and would not be suggestive of **sending a query to a resource manager, wherein said resource manager manages information corresponding to a resource, said resource manager complying with a common standard for resource managers within the network** as required by claim **2** because, according to Sofman, only one resource manager exists, no common standard exists or is described, and no common standard for resource managers is suggested by Sofman due to the necessity of maintaining only one resource manager with which to view the network snapshot.

b. Sofman cannot teach **managing data stored in memory and organized in table format using said query, including manipulating the data to reflect the current resource state** as required by claim **2.**

Referring to the discussion of Sofman's invention in paragraph Ia above, data granulator **104** and not rehoming optimizer **108** or prepared network database **106** manages the data in the memory. "Data Granulator 104 which accesses network data 102 and builds switch and RCG data into a Database 106, and a Rehoming Optimizer 108 which uses database 106 as input. Rehoming optimizer 108 interfaces with a user through an EUI" (see col. 4, lines 19 - 23). Apparently only data granulator **104** manipulates the data in prepared network database **106**, while the functionality of rehoming optimizer **108** is restricted to merely accessing the data in memory. Sofman cannot teach **...managing data stored in memory and organized in table format using said query** because (1) rehoming optimizer **108** does not manage ANY data in the memory whatsoever, and (2) although data granulator **104** manages data in prepared network

database **106**, data granulator **104** never processes any queries from rehoming optimizer **108**, nor can it according to the independent relationship (between data granulator **104** and rehoming optimizer **108**) described by Sofman. Thus, Sofman cannot teach **...managing data stored in memory and organized in table format using said query, including manipulating the data to reflect the current resource state** as required by claim **2**, because Sofman teaches that rehoming optimizer **108** retrieves only data from prepared network database **106** and does not manage ANY data in the memory, and alternatively data granulator **104** never processes any queries from rehoming optimizer **108** and therefore cannot use a user query for managing data.

Sofman is simply not concerned with providing a generic API type for all resources that is standardized between all resource managers as required by the present claims. Furthermore, querying and management is not standardized based on the resource managers as required by the claims but based on the resource understanding the API as is common in the prior art. Sofman's concept of resource management is based, at best, entirely on prior art techniques using resource APIs and not based on generic resource manager APIs or any notion of standardization for queries and management between all resources managers as required by the claims.

As Sofman does not teach or suggest either **... sending a query to a resource manager, wherein said resource manager manages information corresponding to a resource, said resource manager complying with a common standard for resource managers within the network** or **...managing data stored in memory and organized in table format using said query, including manipulating the data to reflect the current resource state**. It is respectfully urged that the Examiner's rejection of claims **2, 6** and **7** should not be sustained.

III. The rejection of claims 3, 8, 10, 11, 15, 16 and **18 under U.S.C. § 102 is incorrect.**

In addition to the forgoing:

a. Sofman does not teach a **plurality of application program interface means for enabling said processor which is connected to a memory, to provide an interface between one or more resource requestors and data organized in a plurality of tables, each of said plurality of tables corresponding to one of a plurality of resources** as required by claim **3**.

Initially, Sofman does not describe an API of any type much less a plurality of APIs **providing an interface between** any **requestor** and any **data organized in a plurality of tables**. Moreover, Sofman does not describe a plurality of APIs **providing an interface**

between any **requestor** and **tables corresponding to one of a plurality of resources.** Sofman is simply unconcerned with standardizing APIs based on a resource manager or resource management system of any type. Sofman's system apparently makes use of prior art resource-based APIs as Sofman teaches or discloses no more.

Here again, *although the rejection alleged by the Examiner is not specific*, assuming that the user somehow queries prepared network database **106** via rehoming optimizer **108**, then data granulator **104** is never included in the query process. With regard to rehoming optimizer **108**, Sofman never describes a **plurality of application program interface means for enabling said processor which is connected to a memory, to provide an interface between one or more resource requestors and data organized in a plurality of tables, each of said plurality of tables corresponding to one of a plurality of resources**, but instead merely describes using EUI driver **114** to allow users to access rehoming optimizer **108**. EUI driver **114** cannot interface with prepared network database **106**, but instead, apparently the internal functionality of rehoming optimizer **108** access the data in prepared network database **106.** Therefore, at best, any APIs used by rehoming optimizer **108** must interface between prepared network database **106** and rehoming optimizer **108**, and not the user-requestor. Thus, Sofman cannot disclose an **interface between one or more resource requestors and data organized in a plurality of tables** as required by claim **3**.

Nowhere does Sofman describe any **application program interface means**, whatsoever, much less one that **provides an interface between one or more resource requestors and data organized in a plurality of tables**. Thus, Sofman cannot teach **a plurality of application program interface means for enabling said processor which is connected to a memory, to provide an interface between one or more resource requestors and data organized in a plurality of tables, each of said plurality of tables corresponding to one of a plurality of resources** as required by claim **3**.

b. Furthermore, Sofman does not teach a **plurality of application program interface means** (providing an interface between one or more resource requestors and data organized in a plurality of tables) wherein **each of said plurality of application program interface means having a sending means for sending a query; and a managing means for managing data stored in said memory and organized in table format using said query; wherein said**

application program interface means provides system-wide interface with said data as required by claim **3**.

Here again, no matter how Sofman is interpreted, Sofman does not teach an API that provides an interface to a resource requestor and data, where the API contains both a **sending means for sending a query** and a **managing means for managing data stored in said memory and organized in table format using said query**. As discussed several times above, Sofman describes no more than a prior art resource API, of which Appellant is well aware and has described in the background section. Sofman does not teach or suggest a resource manager API as defined in claim **3**. If any API is used by Sofman at all, the API is a resource-based prior art type and therefore could NOT **manage data stored in said memory**, but instead merely retrieves data from the resources memory.

Specifically with regard to rehoming optimizer **108**, Sofman simply does not describe rehoming optimizer **108** as having any mechanism for **managing any data** whatsoever **stored** in network database **106**. Therefore, Sofman certainly cannot teach rehoming optimizer **108** as having an API with a **managing means for managing data stored in said memory and organized in table format using said query** as described above, and thus cannot teach or suggest the limitations of claim **3**.

Furthermore, while data granulator **104** may manage the data in prepared network database **106**, Sofman does not describe data granulator **104** as using an API for this process. Moreover, if such an API exists, it would only be a prior art resource API and therefore could not provide **provides system-wide interface with said data** as required by claim **3**.

Additionally, Sofman does not teach that any API associated with data granulator **104** provides an interface between requestor and data table corresponding to one of a plurality of resources, and which further contains a **means for sending a query** and a **means for managing data stored in said memory and organized in table format using said query** as particularly required by claim **3** Instead, at best data granulator **104** utilizes prior art resource-based API for retrieving data from network data **102**. Then, functionality specific to data granulator **104** manages the data within prepared network database **106** and apparently not any resource API that might exist.

Because Sofman does not teach or suggest a **plurality of application program interface means for enabling said processor which is connected to a memory, to provide an**

interface between one or more resource requestors and data organized in a plurality of tables, each of said plurality of tables corresponding to one of a plurality of resources or **each of said plurality of application program interface means having a sending means for sending a query; and a managing means for managing data stored in said memory and organized in table format using said query; wherein said application program interface means provides system-wide interface with said data**, the Examiner has not met the burden, and it is respectfully urged that the Examiner's rejection of claims **3, 8, 10, 11, 15, 16** and **18** should not be sustained.

IV. The rejection of claim 5 under U.S.C. § 102 is incorrect.

In addition to the forgoing:

Sofman does not teach, describe or in any way suggest a resource management means with **table manager resource manager; a queue logic manager resource manager; a system manager resource manager; and a shared memory manager resource manager**.

Sofman simply does not teach or suggest the concept of a resource management means with multiple resource managers. At best, Sofman suggests that a single resource manager might be possible when rehoming optimizer **108**, prepared network database **106** and data granulator **104** are taken as a whole. There is no suggestion whatsoever of multiple resource managers contained within a resource management means.

Because Sofman does not teach or suggest a resource management means with **table manager resource manager; a queue logic manager resource manager; a system manager resource manager; and a shared memory manager resource manager**, the Examiner has not met the burden and it is respectfully urged that the Examiner's rejection of claim **5** should not be sustained.

V. The rejection of claim 9 under U.S.C. § 103 is incorrect.

The Examiner has rejected claim **9** under 35 U.S.C. § 103(a) as being unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Taylor, et al. (U.S. Patent No. 5,912,961).

Here, the Examiner contends that one of ordinary skill in the art would modify the rehome optimization method taught by Solman with the API suspend command taught by

Taylor. However, Solman specifically solves the problem of data collisions in database **106** by synchronizing data granualtor **104** with rehome optimizer **110**, and in so doing, data granualtor **104** with rehome optimizer **110** can work independently and a fresh snapshot of the network is always available for rehome optimizer **110** (col. 4, lines 30 - 40). Simply stated, the ordinary artisan would not be motivated to reach the invention recited in claim **9** without using the present specification as a roadmap or template because Solman solves the problem in a different manner.

"[I]t is impermissible to use the claimed invention as an instruction manual or `template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious .... This court has previously stated that `[o]ne cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention.' " *In re Fritch*, 972 F.2d 1260, 23 USPQ2d 1780 (Fed. Cir. 1992). Obviousness can not be established by hindsight combination to produce the claimed invention. ... [I]t is the prior art itself, and not the applicant's achievement, that must establish the obviousness of the combination. *'In re Dance*, 160 F.3d 1339, 48 USPQ2d 1635 (Fed. Cir. 1998). Close adherence to this methodology is especially important in the case of less technologically complex inventions, where the very ease with which the invention can be understood may prompt one `to fall victim to the insidious effect of a hindsight syndrome wherein that which only the inventor taught is used against its teacher.' "); *Al-Site Corp. v. VSI International Inc.*, 174 F.3d 1308, 1324, 50 USPQ2d 1161, 1171 (Fed. Cir. 1999).

Therefore, because the Examiner has not met the burden, and it is respectfully urged that the Examiner's rejection of claim **9** should not be sustained.

VI. The rejection of claims 12 - 14 and 17 under U.S.C. § 103 is incorrect.

The Examiner has rejected claim nos. **12-14** and **17** under 35 U.S.C. § 103(a) as being unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Gottlieb (U.S. Patent No. 5,920,621).

Here again the Examiner contends that one of ordinary skill in the art would modify the rehome optimization method taught by Solman with an agent entry via an API and receiving heartbeat messages according to the teaching of Gottlieb. There is simply no motivation for such

a modification suggested by either Solman or Gottlieb. The mere fact that the prior art could be readily modified to arrive at the claimed invention does not render the claimed invention obvious; the prior art must suggest the desirability of such a modification. *In re Ochiai*, 71 F.3d 1565, 1570, 37 U.S.P.Q.2d 1127, 1131 (Fed. Cir. 1996); *In re Gordon*, 733 F.2d 900, 903, 221 U.S.P.Q. 1125, 1127 (Fed. Cir. 1984).

Therefore, because the Examiner has not met the burden and it is respectfully urged that the Examiner's rejection of claims **12-14** and **17** under 35 U.S.C. § 103(a) should not be sustained.

VII. The rejection of claims 12 - 14 and 17 under U.S.C. § 103 is incorrect.

The Examiner has rejected claim **20** under 35 U.S.C. § 103(a) as being unpatentable over Sofman (U.S. Patent No. 5,937,042) in view of Reto et al. (U.S. Patent No. 5,825,857).

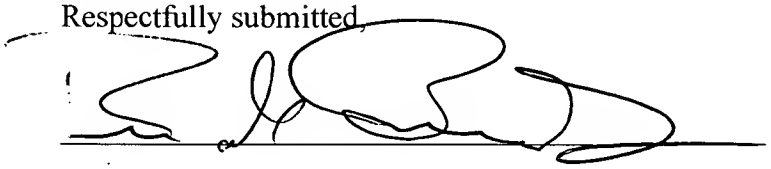
Appellant's representative admits that it might again be possible to modify Solman. However, it is unclear why an artisan would look to Reto for an improvement. Solman and Reto are not analogous art in that a person with ordinary skill in the art would not be expected to examine Reto because Reto is neither telecom resource management art nor does he provide a solution to the present problem. Therefore, at a minimum, the references themselves must state a reason for combining the teachings of the references to reach the presently-claimed invention. As the Examiner has used generally available motivation as a reason for suggesting the combination of references, it is respectfully submitted that the rejection is improper.

Therefore, because the Examiner has not met the burden, it is respectfully urged that the Examiner's rejection of claim **20** under 35 U.S.C. § 103(a) has been overcome.

VIII. Conclusion

In view of the above comments, it is respectfully urged that the Examiner's rejection of the claims should not be sustained.

Respectfully submitted,



Rudolph J. Buchel Jr.
Reg. No. 43,448
Jones, Day, Reavis & Pogue
P.O. Box 660623
Dallas TX 75266
214 969-2990
214 969-5600 Fax

Attorney for Appellant

IX. APPENDIX OF CLAIMS (37 C.F.R. 1.192(c)(9))

The text of the claims involved in the appeal are:

1. A computer in a telecommunications network, comprising:

a processor; and

a resource management means for enabling said processor to provide standardized management of multiple resources including internal operational resources, external components, and applications processing data, wherein said resource management means comprises one or more resource managers, said resource managers being one of:

a semaphore resource manager;

a switch controller resource manager;

an agent resource manager;

a call data block resource manager;

a service logic resource manager; or

a switch resource resource manager;

wherein each of said resource managers comprises:

one or more resource manager application program interfaces that manage said internal operational resources, said external components, and said applications processing data; and

one or more data storing means for enabling said processor to store data in table format related to said internal operational resources, said external components, and said applications processing data, said application interfaces manipulating the data to reflect the current resource state.

2. A method for managing resources within a network, comprising:

(i) sending a query to a resource manager, wherein said resource manager manages information corresponding to a resource, said resource manager complying with a common standard for resource managers within the network; and

(ii) managing data stored in memory and organized in table format using said query, including manipulating the data to reflect the current resource state;

wherein said data is one of:

semaphore data;

switch controller data;

agent data;

call data block data;

service logic program data; or

switch data.

3. A computer in a telecommunications network, comprising:

a processor; and

plurality of application program interface means for enabling said processor which is connected to a memory, to provide an interface between one or more resource requestors and data organized in a plurality of tables, each of said plurality of tables corresponding to one of a plurality of resources, each of said plurality of application program interface means comprising:

sending means for sending a query; and

managing means for managing data stored in said memory and organized in table format using said query; wherein said application program interface means provides system-wide interface with said data;

wherein each of said plurality of application program interface means complies with a common standard for application program interfaces;

wherein each of said plurality of application program interface means manipulating the data to reflect the current resource state.

4. The computer of claim 1, wherein said data within said data storing means comprises one of:

switch controller data;
call data block data; or
service logic program data.

5. The Computer of claim 1, wherein said resource management means further comprises:

a table manager resource manager;
a queue logic manager resource manager;
a system manager resource manager; and
a shared memory manager resource manager.

6. The method of claim 2, wherein step (ii) comprises the step of:
updating said data stored in said memory and organized in table format using said query.

7. The method of claim 2, wherein step (ii) comprises the step of:
retrieving said data stored in memory and organized in table format using said query.

8. The computer of claim 3, wherein said data comprises one or more of:
semaphore data;
switch controller data;
agent data;
call data block data;
service logic program data; and
switch data.

9. The computer of claim 3, wherein one of said application programmer interface means is one of:

create table semaphore;
initialize table semaphore;
create semaphore;
initialize semaphore;
delete semaphore;
attach semaphore;
lock semaphore table;
unlock semaphore table;
lock semaphore table entry;
unlock semaphore table entry;
lock semaphore one entry;
recover table semaphore;
get semaphore size;
get table semaphore value; or
print semaphore.

10. The computer of claim 3, wherein one of said application programmer interface means is one of:

create switch controller common library memory segment;
delete switch controller common library memory segment;
attach switch controller common library memory segment; and
detach switch controller common library memory segment.

11. The computer of claim 3, wherein one of said application programmer interface means is one of:

set up an operational measurements IPC;
update an operation measurements IPC;
print an operational measurements IPC;
get an operational measurements attribute; and
set an operational measurements attribute.

12. The computer of claim 3, wherein one of said application programmer interface means is one of;

get time;
create a heartbeat table;
delete a heartbeat table;
attach to a heartbeat table;
detach from a heartbeat table;
create a heartbeat entry;
delete a heartbeat entry;
get a heartbeat handle;
request heartbeat;
respond heartbeat;
set heartbeat interval;
get heartbeat attributes; and
print heartbeat table.

13. The computer of claim 3, wherein one of said application programmer interface means is one of:

create agent segment;
delete agent segment;
attach agent segment; and
detach agent segment.

14. The computer of claim 3, wherein one of said application programmer interface means is one of:

create agent entry;
delete agent entry;
update agent state;
agent select;
agent destination number to terminal identifier conversion;
get agent data;
set agent data;
get agent attribute;
set agent attribute;
get agent handle;
get agent counts;
print agent table;
print agent entry; and
print agent search table.

15. The computer of claim 3, wherein one of said application programmer interface means is one of:

create group entry;
delete group entry;
get group handle;
get group data;
set group data;
increase calls queued on group;
decrease calls queued on group;
get group count;
print group table;
print group entry; and
print group search table.

16. The computer of claim 3, wherein one of said application programmer interface means is one of:

create assign entry;
delete assign entry by keys;
delete agent assign;
delete group assign;
get assign by keys;
get assign count;
get agent assign count;
get group assign count; and
print assign table.

17. The computer of claim 3, where one of said application programmer interface means is one of:

create call data block table;
delete call data block table;
attach call data block table;
detach call data block table;
create call data block entry;
delete call data block entry;
call data block search call identifier by port identifiers;
get call data block data;
set call data block data;
print call data block data;
return call data block attribute;
set call data block attribute;
get number call data block entries;
print call data block table; or
print call data block entry.

18. The computer of claim 3, wherein one of said application programmer interface means is one of:

create service logic program table;
delete service logic program table;
attach service logic program table;
detach service logic program table;
create service logic program entry;
delete service logic program entry;
get service logic program data;
set service logic program data;
print service logic program data;
get service logic program attribute;
set service logic program attribute;
service logic program search call identifier by terminal identifier;
get service logic program count;
print service logic program table; or
print service logic program entry.

19. The computer of claim 4, wherein said data comprises one or more of:

switch controller IPC data;
switch controller CPU availability data;
switch controller disk availability data;
agent operational measurement count data, wherein said agent operational measurement count data is data collected for one or more agents controlled a switch controller;
a switch port operational measurement count data, wherein said switch port operational measurement count data collected for one or more switches controlled by said switch controller;
control table data for control tables within said switch controller; or
heartbeat data for heartbeating of routines within said switch controller.

20. The computer of claim 4, wherein said data stored in said table format comprises:

call identifying information;

calling number;

called number;

call leg information; and

billing time point information.